Una Revisión Comparativa de Métodos Híbridos para Análisis de Sentimientos Multiclase

A Comparative Review of Hybrid Methods for Multiclass Sentiment Analysis

Brian Keith & Claudio Meneses
brian.keith@ucn.cl, cmeneses@ucn.cl
Department of Computing and Systems Engineering
Universidad Católica del Norte, Antofagasta, Chile

RESUMEN

El análisis de sentimientos es un área de la minería de datos con potencialmente muchos dominios de aplicación, pero a la vez con varios desafíos de investigación. Los métodos tradicionales aplicados hasta ahora provienen desde el área de aprendizaje automático y de procesamiento del lenguaje natural principalmente. Uno de los desafíos actuales es el escalar resultados obtenidos en la clasificación binaria de sentimientos hacia una clasificación multiclase, donde usualmente los resultados obtenidos sufren una degradación importante. Este trabajo resume el estado del arte respecto a los principales métodos aplicados para el análisis de sentimientos desde texto en lenguaje natural. Se revisan técnicas de clasificación multiclase en general, métodos de análisis de sentimientos y representaciones de texto para el mismo propósito, concluyendo con una revisión de las técnicas de aprendizaje profundo para el procesamiento del lenguaje natural. La discusión se centra en el análisis del estado del arte para responder preguntas de investigación planteadas inicialmente, y cuyas respuestas son fundamental para guiar el diseño de un método híbrido que obtenga resultados competitivos con respecto a una línea base derivada desde el estado del arte y que mejore sustancialmente los resultados para un dominio especifico, para el caso del análisis de sentimientos multiclase. En las conclusiones y trabajo futuro se plantean algunas alternativas de diseño del método híbrido propuesto como principal objetivo de este trabajo de investigación.

Palabras clave: Minería de opiniones, análisis de sentimientos multiclase, aprendizaje automático, representación de texto, aprendizaje profundo.

ABSTRACT

Sentiment analysis is an area of data mining with potentially many application domains, but also with several research challenges. Traditional methods applied so far come from the area of machine learning and processing of natural language mainly. One of the current challenges is how to scale results gotten in binary sentimental classification towards a multi-class classification where the results usually suffer significant degradation. This paper summarizes the state of the art regarding the main methods used for sentimental analysis from text written in natural language. Multiclass classification techniques, methods for sentiment analysis, text representation techniques and deep learning for natural language processing are reviewed. The discussion focuses on the analysis of the state of art to answer research questions initially raised, and whose answers are essential to guide the design of a hybrid method to obtain competitive results with respect to a baseline derived from the state of the art and substantially improve outcomes for a specific domain, in the case of multiclass sentimental analysis. The conclusions and future work discuss some alternative designs for the hybrid method proposed as the main objective of this research work.

Keywords: Opinion mining, multiclass sentimental analysis, machine learning, text representation, deep learning.

INTRODUCCIÓN

Este artículo corresponde al análisis sistemático de los métodos aplicados en el estado del arte en el problema de identificar y clasificar sentimientos a partir de texto en lenguaje natural. En esta revisión sistemática del estado del arte se analizan los distintos enfoques utilizados, y se realiza una evaluación cualitativa de los métodos del estado del arte.

El objetivo primordial de este análisis es guiar el diseño de un método híbrido y/o recursivo, que permita mejorar los resultados actuales del estado del arte en lo referente al análisis de sentimientos multiclase. La revisión sistemática se orienta a discutir una serie de preguntas respecto a los métodos y técnicas más relevantes utilizadas para realizar análisis de sentimientos y clasificación. La Tabla 1 resume las preguntas que dirigen el desarrollo de esta revisión sistemática.

Tabla 1: Preguntas que guían la revisión del estado del arte.

Categorías	Preguntas
Clasificación multiclase	¿Qué técnicas de clasificación multiclase existen en general?
	¿Es posible aprovechar la estructura ordinal de las clases en este caso?
	¿Cuál es el método más adecuado para la determinación de polaridad multiclase?
Métodos de clasificación de sentimientos	¿Qué métodos no han sido aplicados exhaustivamente en la tarea de determinación de polaridad?
	¿Por qué estos métodos podrían ser útiles?
Representación de texto	¿Qué tipos de representación del texto es la más adecuada para este problema?
	¿Qué técnicas de preprocesamiento son necesarias para obtener esta representación?
	¿Qué información semántico-sintáctica almacenan estas representaciones?
Aprendizaje profundo	¿Qué técnicas de clasificación basada en métodos recursivos profundos han sido aplicadas con éxito en determinación de polaridad?
	¿Qué ventajas y desventajas poseen las diferentes técnicas?
	¿Qué falencias presentan en el caso multiclase?

El resto del artículo se organiza de la siguiente forma. Primero se revisan las distintas técnicas de clasificación multiclase existentes. En segundo término, distintos métodos de clasificación de sentimientos utilizados hoy en día. En tercer término, las técnicas más comunes para representar texto. En cuarto lugar, las técnicas de aprendizaje profundo para el procesamiento del lenguaje natural. Finalmente, se discuten las posibles respuestas a las preguntas planteadas en la Tabla 1, y se enuncian las conclusiones en función de lo requerido para el diseño de un método de análisis de sentimientos multiclase, siguiendo un enfoque de aprendizaje híbrido.

CLASIFICACIÓN MULTICLASE

La mayor parte de esta sección está basada en el trabajo de revisión de Lorena et al. 2008. Muchas de las técnicas de aprendizaje automático han sido diseñadas originalmente para problemas de clasificación binaria. No obstante, una multitud de aplicaciones requiere una mayor granularidad en clasificación. Por lo que se hace necesario disponer de métodos que permitan categorizar los datos en más de dos clases. Algunos enfoques pueden realizar esto de manera natural (e.g., clasificador simple de Bayes), no obstante, existen otros que no tienen una generalización natural que garantice un buen rendimiento o un entrenamiento eficiente (e.g., máquinas de soporte vectorial).

Los problemas de clasificación multiclase son intrínsecamente más complejos que los problemas de clasificación binaria, pues el clasificador generado debe ser capaz de separar los datos en más categorías, lo que incrementa las probabilidades de errores de clasificación. En la literatura se han encontrado dos enfoques para tratar esta problemática. El primero consiste en adaptar la operación interna del algoritmo de entrenamiento del clasificador, y el segundo en descomponer el problema multiclase en un conjunto de problemas de dos clases.

La extensión de un clasificador a un modelo multiclase siguiendo el primer esquema tiene una serie de complicaciones, siendo poco práctico y costoso en algunos casos. Por lo tanto, la estrategia más común es la descomposición del problema en varios subproblemas de clasificación binaria. Además, existen una serie de ventajas al utilizar este enfoque, que incluso benefician a los métodos que se generalizan directamente al caso multiclase. Por otra parte, también permite paralelizar el

trabajo, debido a que cada problema se puede abordar por separado.

Métodos tradicionales

El enfoque de separación en subproblemas binarios consta, a grandes rasgos, de dos pasos: descomposición y agregación. El primero consiste en separar el problema original y determinar los clasificadores binarios a utilizar. El segundo consiste en determinar cómo se combinarán los resultados obtenidos por cada clasificador binario. Se pueden además distinguir las estrategias tradicionales y las estrategias jerárquicas. Las primeras se detallan en la Tabla 2, donde *k* representa la cantidad de clases en el problema.

Tabla 2: Métodos de clasificación multiclase tradicionales.

Enfoque	Descripción
OAA (1-vs-all)	Se entrena un clasificador por cada clase contra todas las otras. El método presenta desventajas si el número de instancias es muy bajo para una clase. Para la agregación se suele escoger la clase que entregó la mayor probabilidad de pertenencia. En este caso se requieren *k* clasificadores.
OAO (1-vs-1)	Se entrena un clasificador por cada par de clases *i* y *j*, este clasificador se utiliza para distinguir si el ejemplo pertenece a la clase *i* o a la clase *j*. Esto se repite para cada posible combinación. El entrenamiento suele ser rápido pues solo incluyen dos clases. Para la agregación se suele utilizar una votación para determinar la clase correcta. Se escoge la clase que tiene la mayor cantidad de votos. Se requieren $k(k-1)/2$ clasificadores.
ECOC	Se codifican las clases siguiendo un esquema de corrección de errores (Error Correcting Output Codes), es decir, se trata el aprendizaje como un proceso de comunicaciones sujeto a ruido. Se codifican las clases de tal forma de tener redundancia y resistencia al ruido. Para la agregación se utiliza una función de decodificación Hamming para encontrar la clase correcta. Para determinar la cantidad de clasificadores existen varias técnicas (Dietterich & Bariki, 1995).
DA	La descomposición adaptada se centra en buscar la codificación más apropiada para cada problema utilizando diferentes técnicas y heurísticas. Esto hace la fase de descomposición más compleja. Para la agregación se puede utilizar la función de decodificación Hamming. La cantidad de clasificadores es variable y depende del problema en particular.

Como se indicó, es posible también abordar el problema de clasificación multiclase de manera jerárquica. Esto implica disponer de los clasificadores siguiendo un esquema jerárquico, realizando discriminaciones generales primero y sucesivamente refinar éstas hasta encontrar la clase correspondiente. En general, la introducción de una clasificación jerárquica puede reducir la complejidad del problema abordado. En general, la literatura hace distinción entre dos tipos de clasificadores jerárquicos:

1. Basados en grafos acíclicos dirigidos (DAG, por sus siglas en inglés).
2. Basados en árboles binarios dirigidos (DBT, por sus siglas en inglés).

Notar que estos últimos son un caso particular de los primeros, pero por simplicidad de tratamiento se analizan por separado. Debido a la gran cantidad de posibles jerarquías que se pueden formar, existe una serie de estrategias y heurísticas diferentes que se pueden aplicar para generar la topología general del grafo (Lorena et al. 2009)

Dentro del contexto de la minería de opiniones una de las aplicaciones de la clasificación jerárquica para la determinación de polaridad es la siguiente: determinar primero si un texto expresa o no un sentimiento (es decir, neutral vs. todas) y luego en el siguiente nivel jerárquico clasificar en negativo o positivo según corresponda. Esta estrategia ha sido utilizada con éxito previamente.

No obstante, se debe notar que hasta ahora se ha abordado el problema de clasificación multiclase en general. Sin tener en consideración la naturaleza propia del dominio en que se está trabajando. Específicamente, la detección de polaridad de sentimientos con múltiples clases tiene una diferencia con respecto a un problema multiclase cualquiera: las clases son ordinales.

Métodos de clasificación ordinal

Los métodos estudiados hasta el momento se centran todos en datos nominales, es decir, en los que las etiquetas clases pertenecen a un conjunto sin un orden natural. En contraste, se tiene el problema de clasificación ordinal (a veces llamada regresión ordinal), que se encuentra en

un punto medio entre la clasificación clásica y la regresión (Wang et al 2014). En la literatura se encuentran tres enfoques para tratar los problemas de clasificación ordinal (Kotsiantis y Pintelas 2004):

1. Trabajar con la escala como si fuese un problema de clasificación multiclase común.
2. Utilizar regresión para estimar un valor continuo y luego discretizarlo.
3. Algoritmos especializados para clasificación ordinal.

El primer enfoque no aprovecha la estructura interna de los datos al omitir el orden natural existente en las clases. Si bien existen varios ejemplos de métodos que funcionan exitosamente siguiendo este esquema, se esperaría que contar con la información del orden que sigue cada clase, se pudiese obtener una clasificación más precisa mediante la explotación de esta estructura.

El segundo enfoque tiene como problema que es una solución ad hoc para un problema de clasificación ordinal, pues los modelos de regresión han sido pensados para datos continuos, y si bien es posible llevar los datos ordinales a una escala real y luego ejecutar un paso de post-procesamiento para obtener las clases, esto no es ideal.

Los terceros métodos aprovechan la estructura de clasificación ordinal. Esto puede hacerse mediante el uso de un algoritmo de aprendizaje automático modificado para explotar esta estructura, no obstante, algunos de ellos presentan algunas complejidades en cuanto a implementación y entrenamiento. Existen otros enfoques más simples que ofrecen resultados prometedores sin incurrir en una complejidad computacional mayor. Estos consisten en aplicar descomposiciones del problema de una forma específica o modificaciones a la función objetivo durante el entrenamiento de los métodos.

Es importante considerar el hecho de que la diferencia entre clasificación ordinal y clasificación nominal no es notoria en el caso de clasificación binaria, pues siempre existe un orden implícito en cuanto a "clase positiva" y "clase negativa". Dado que la mayoría de los trabajos de determinación de polaridad todavía se centran en el caso binario, es natural que no se exploren en detalle los métodos de clasificación ordinal. Se detallarán a continuación algunos de los métodos especializados que pueden ser utilizados en clasificación ordinal. No se detallarán los métodos basados en clasificación nominal ni de regresión.

La clasificación ordinal no busca maximizar solo la precisión de clasificación, sino también minimizar las distancias entre la clase predicha y la verdadera. Esto generará un cierto sesgo hacia la clase intermedia.

Un enfoque que permite incluir estas consideraciones requiere de la definición de una matriz de costos y de una función de riesgo condicional de cada decisión. La idea de esta función es que el riesgo de elegir la clase i viene dado por los costos de clasificación incorrecta definidos en la matriz y la incertidumbre en el conocimiento acerca de la verdadera clase de la instancia, expresado mediante probabilidades a posteriori. El objetivo es entonces minimizar el costo de clasificación errónea, esto corresponde a escoger la clase con el menor riesgo condicional. Esta función de riesgo se puede incorporar a las funciones objetivo utilizadas en el entrenamiento de métodos convencionales de clasificación nominal (Kotsiantis y Pintelas 2004).

Otro enfoque simple que se puede utilizar no está basado en costos, sino que primero transforma el problema de k-clases ordinales en $k-1$ problemas de clasificación binaria, donde el objetivo es diferenciar si la instancia es mayor a un cierto valor V_i. En función de esto se construye una estimación de las probabilidades de cada instancia, la ordinalidad del problema permite aprovechar la información entregada por los distintos clasificadores (Frank y Hall 2001).

Estos enfoques se pueden aplicar complementariamente con los modelos de clasificación nominales para obtener mejoras (ya sea mediante la separación del problema en subproblemas o mediante la modificación de la función objetivo a optimizar). En cuanto a los enfoques que utilizan algoritmos de aprendizaje automáticos especializados, estos son más complejos, y requieren de cambios no triviales en los métodos de entrenamiento.

En el campo del análisis de sentimientos, se han utilizado estrategias que penalizan en función de la distancia de la clase asignada con respecto a la clase verdadera durante el entrenamiento. La calidad del modelo dependerá de la función de distancia utilizada (Pang y Lee 2005).

CLASIFICACIÓN DE SENTIMIENTOS

La tarea particular que se aborda en este trabajo es la determinación de polaridad (e.g., negativo, positivo o neutral). Según Ravi y Ravi (2015) se distinguen tres enfoques para realizar esta tarea: basados en aprendizaje automático, basados en léxicos y métodos híbridos.

Entre los enfoques basados en aprendizaje automático los más utilizados son las máquinas de soporte vectorial, el clasificador simple de Bayes y las redes neuronales. En cuanto a los métodos basados en léxicos se destaca el uso de diccionarios de opinión y de ontologías tales como SentiWordNet (Baccianella et al 2010).

En el contexto del análisis de sentimientos se define como método híbrido aquel que combina los dos enfoques tradicionales, es decir, utiliza tanto técnicas de aprendizaje automático como técnicas basadas en la semántica del texto, haciendo uso de diccionarios de sentimientos. Debido a la mayor complejidad computacional de los métodos híbridos no se utilizan tan frecuentemente como sus contrapartes tradicionales, aunque recientemente existe una tendencia a desarrollar nuevos enfoques híbridos (Medhat et al 2014, Ravi y Ravi 2015).

Sin contar los métodos clásicos como SVM, NB, redes neuronales y enfoques basados en léxico, la revisión de la literatura muestra que hay varios métodos que no han sido explotados de manera exhaustiva. Se listan los principales a continuación: reglas de asociación, lógica difusa, random forests, campos aleatorios condicionales (CRF) y ontologías.

El uso de reglas de asociación puede facilitar el descubrimiento de palabras de opinión utilizadas en conjunto. Por otra parte, como el sentimiento generalmente se expresa de manera vaga, la lógica difusa es claramente un modelo apropiado para abordar este problema de una manera más robusta. Los CRF pueden complementarse con información del dominio para obtener una mejor extracción de aspectos. Las ontologías pueden ser útiles para generalizar las medidas estándar de sentimientos. En particular, el complementar los métodos de aprendizaje automático con una ontología de manera adecuada para el análisis de sentimientos es un campo de estudio aun abierto. Éstas se podrían utilizar para resolver problemas de escalabilidad y ambigüedad en los métodos actuales (Ravi y Ravi 2015). Se detallarán a continuación brevemente los métodos que no han sido explotados exhaustivamente, con el propósito de visualizar posibles nuevas aplicaciones.

Reglas de asociación

El aprendizaje de reglas de asociación es un método para descubrir patrones y relaciones interesantes entre variables dentro de conjuntos de datos. Su objetivo es identificar reglas interesantes de la forma $X \rightarrow Y$, donde X e Y son conjuntos de variables, mediante la evaluación de un conjunto de métricas de evaluación (Agrawal et al 1993). En principio, las reglas de asociación no construyen un clasificador, sino que buscan detectar patrones de co-ocurrencia. No obstante, existen enfoques que plantean el uso de reglas de asociación para realizar la tarea de clasificación (Liu et al. 1998).

En general, dentro del campo de minería de opiniones, las reglas de asociación son utilizadas para obtener los aspectos más importantes de una entidad o para la generación de léxicos de opinión. No se han encontrado aplicaciones directas de este método a la determinación de polaridad. Esto se puede deber al hecho que las reglas de asociación no entregan directamente un clasificador, por lo que su aplicación en la tarea de determinación de polaridad sería más bien complementaria (Ravi & Ravi 2015).

Lógica difusa

La lógica difusa es una forma de lógica multivaluada en la cual los valores de verdad de las variables pueden ser cualquier número real entre 0 y 1. En contraste con la lógica Booleana, donde las variables solo pueden tomar valores 0 o 1. La lógica difusa ha sido aplicada para manejar el concepto de verdad parcial, donde un valor de verdad puede encontrarse en un punto intermedio entre completamente verdadero y completamente

falso. Notar que la lógica difusa difiere de la teoría de probabilidades pues no busca modelar formalmente la incertidumbre, sino la ambigüedad (Kilr y Yuan 1995).

A modo de ejemplo se muestra en la Figura 1 una comparación de la perspectiva de lógica clásica (derecha) y de lógica difusa (izquierda) para el concepto de personas "altas" y personas "no altas". En el caso de la lógica clásica se observa que existe una frontera bien definida entre aquellas personas que clasifican como altas y aquellas que no clasifican como altas. En cambio, en la visión de lógica difusa se parte de la idea de que no existe una frontera bien definida para definir si una persona es alta o no, sino que en cambio existe una variación continua de la pertenencia al conjunto. La lógica difusa es capaz de modelar expresiones vagas o ambiguas que la lógica clásica no podría abordar.

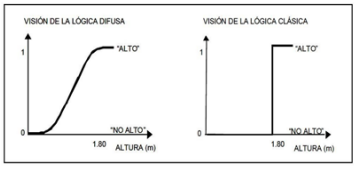

Figura 1: Comparación entre lógica difusa y lógica clásica.

En el campo del análisis de sentimientos la lógica difusa permite abordar las problemáticas de ambigüedad en el texto. Si bien existen trabajos que aplican lógica difusa, este enfoque no ha sido explotado exhaustivamente. Uno de los trabajos que cabe destacar es el realizado por Li y Tsai (2013), quienes han desarrollado un framework de clasificación basado en análisis difuso de conceptos formales. Quienes consideran el desarrollo de un clasificador multiclase un aspecto importante a trabajar en el futuro. En general, los modelos difusos pueden ser utilizados para abordar los aspectos ambiguos en el procesamiento de lenguaje natural (Ravi y Ravi 2015).

Random forests

Los bosques aleatorios (RF, por sus siglas en inglés) son una técnica general de árboles de decisión aleatorios. El método combina la idea de bagging con la selección aleatoria de características, con el propósito de construir árboles de decisiones con varianza controlada (Ho 1995, Ho 1998). Es un método combinado (ensemble method) para tareas de clasificación y regresión, que opera mediante la construcción de múltiples árboles de decisión durante el entrenamiento. Se generan varios árboles utilizando estos subconjuntos aleatorios y luego se combina el resultado de estos árboles independientes de tal forma de obtener el resultado final, como se aprecia en la Figura 2.

En el caso de clasificación la clase determinada corresponde a la moda de las clases entregadas por cada árbol. En el caso de regresión corresponde a la predicción promedio de los árboles individuales. Los árboles de decisión aleatorios corrigen la tendencia de los árboles de decisión de sobre ajustarse a su conjunto de entrenamiento (Friedman et al. 2001).

En el área de minería de opiniones, los RF han sido aplicados con éxito en tareas de clasificación y predicción, superando consistentemente al enfoque más comunes utilizados (SVM y NB) en todos los casos estudiados. Si bien ha habido estudios sobre esta técnica, no ha sido completamente explorada. El éxito de esta técnica indica un alto potencial de aplicabilidad en este trabajo (Ravi y Ravi 2015).

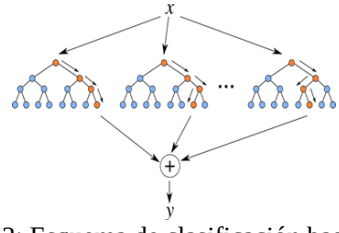

Figura 2: Esquema de clasificación basada en random forests.

Campos aleatorios condicionales (CRF)

Los campos aleatorios condicionales son un tipo de método de modelamiento estadístico usualmente aplicado en reconocimiento de patrones y aprendizaje automático, donde son utilizados para realizar predicciones estructuradas. Normalmente, un clasificador ordinario predice una etiqueta para una muestra sin considerar las muestras circundantes, en cambio, los CRF toman en cuenta el contexto al realizar la clasificación. En particular, los CRF de cadenas lineales han sido utilizados ampliamente en el área de procesamiento de lenguaje natural,

donde permite predecir secuencias de etiquetas para una secuencia de entradas (Sutton y McCallum 2006).

En la Figura 3 se observa la relación entre los modelos CRF y otros modelos estocásticos. En la primera fila se observan modelos que asumen independencia condicional, el más simple de ellos corresponde al clasificador simple de Bayes. Si se considera varias unidades del clasificador simple de Bayes en secuencia se obtiene los modelos ocultos de Markov. Si se generaliza la estructura lineal de los modelos ocultos de Markov a grafos en general se obtiene los modelos generativos dirigidos, correspondientes al modelo más poderoso y complejo entre los que asumen independencia condicional. Al eliminar el supuesto de independencia condicional de cada uno de estos modelos se obtienen los métodos de la segunda fila, correspondientes a regresión logística, las cadenas lineales de CRFs y las CRFs generales, respectivamente. La relación entre estos métodos sigue la misma lógica indicada para la primera (Sutton y McCallum 2006).

En cuanto al análisis de sentimientos han sido aplicados en distintas tareas. Se puede explotar su capacidad de tomar en cuenta el contexto al aplicarse en conjunto con características semánticas y sintácticas, esto permite realizar clasificación de sentimientos, superando a los enfoques tradicionales. Aunque su uso no es muy amplio en cuanto a determinación de polaridad, también se han utilizado con éxito en extracción de aspectos y relaciones, superando incluso a clasificadores que utilizan SVMs multiclase (Ravi y Ravi 2015).

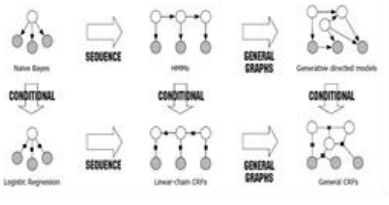

Figura 3: Relación entre clasificadores estocásticos y CRFs.

Un resultado interesante es la formulación de las CRFs como redes neuronales recurrentes (uno de los modelos estudiados en la última sección) propuesto por Zheng et al. (2015). Si bien su trabajo es en el campo de la visión artificial, esta formulación implica que las CRFs pueden ser útiles como base para un método recursivo híbrido para el análisis de sentimientos.

Ontologías

Una ontología es una especificación explicita de una conceptualización. Provee una representación formal del conocimiento, permitiendo razonar sobre este (Gruber 1995). Es mejor que una taxonomía o una base de datos relacional pues captura asociaciones semánticas y relaciones entre conceptos. El campo del análisis de sentimientos busca representar conocimiento útil para el desarrollo de sus tareas mediante ontologías (Ravi & Ravi 2015).

Si bien las principales aplicaciones de las ontologías se encuentran en lo que corresponde a la detección de aspectos (debido a su capacidad de comprender relaciones entre distintos objetos), las ontologías han sido utilizadas con éxito en el área de determinación de polaridad (Ravi & Ravi 2015). El conocimiento almacenado por estas ha sido utilizado para complementar a otros métodos de clasificación de sentimientos, ya sean enfoques tradicionales basados en aprendizaje automático o en léxicos.

Una ontología particularmente útil para el desarrollo de este trabajo es SentiWordNet 3.0, que deriva de WordNet. Esta ontología es un recurso léxico especialmente diseñado para la minería de opiniones. SentiWordNet asigna tres evaluaciones de sentimiento a cada conjunto de sinónimos de WordNet: positividad, negatividad y objetividad.

REPRESENTACIÓN DEL TEXTO

En esta sección se discutirán modelos para representar el texto mediante vectores numéricos. El espacio conformado por estos vectores se denomina un espacio vectorial semántico. Existe una hipótesis subyacente que une a todos los modelos de espacios vectoriales semánticos, llamada la hipótesis de semántica estadística: “Los patrones estadísticos del uso de palabras humano puede ser utilizado para deducir lo que la gente intenta comunicar”. En términos vectoriales, esto significa que, si las unidades de texto tienen vectores similares en una matriz de frecuencia de texto, entonces tienden a tener significados similares (Turney y Pantel 2010).

Esta hipótesis general contiene una serie de sub-hipótesis, descritas en la Tabla 3.

Tabla 3: Sub-hipótesis semánticas utilizadas.

Hipótesis	Descripción
Hipótesis de la bolsa de palabras	Las frecuencias de una palabra en documento tienden a indicar la relevancia del documento a una cierta consulta.
Hipótesis distribucional	Las palabras que ocurren en contextos similares tienden a tener significados similares.
Hipótesis distribucional extendida	Los patrones que co-ocurren con pares similares tienden a tener significados similares.
Hipótesis de la relación latente	Los pares de palabras que co-ocurren en patrones similares tienden a tener relaciones semánticas similares.

Modelos tradicionales

La representación más simple corresponde al enfoque de bolsas de palabra, en la que cada palabra del diccionario está asociada a una posición en el vector. El elemento en esta posición tomará el valor 1 si representa dicha palabra y 0 de otro modo. Esto genera una representación en el espacio $\mathbb{R}^{|V|\times 1}$, donde $|V|$ es el tamaño del diccionario. Se puede ver que esta representación es ineficiente, pues no captura la relación entre las palabras y además genera una representación poco poblada (Liu 2007).

Surge entonces la pregunta intuitiva si es que existe un espacio *k*-dimensional que sea suficiente para codificar todas las semánticas del lenguaje (donde *k* es mucho más pequeño que la cantidad de palabras en el lenguaje). Cada dimensión entonces codificaría algún significado asociado al lenguaje. Por ejemplo, una dimensión semántica podría indicar el tiempo verbal, la cantidad y el género. Un enfoque para realizar esto corresponde al análisis semántico latente, que permite reducir la dimensionalidad mediante la aplicación de una técnica de factorización de matrices (Dumais 2004).

El enfoque convencional es utilizar una descomposición de valores singulares sobre una matriz de co-ocurrencias. Esta matriz puede ser construida mediante varios métodos, como por ejemplo mediante el análisis de las palabras que ocurren de manera conjunta en todos los documentos, otros utilizan una ventana de tamaño fijo alrededor de cada palabra analizada, con el propósito de solo considerar las palabras cercanas (Turney y Pantel 2010).

Si bien estos métodos proveen de vectores útiles para codificar la información semántico-sintáctica de las palabras, vienen asociadas con una serie de problemas. Principalmente, la alta dimensionalidad del problema, el costo de entrenamiento, el hecho de que la matriz es poco poblada, la sensibilidad con respecto a la introducción de nuevas palabras en el diccionario y cambios en el tamaño del corpus (Turney y Pantel 2010).

Métodos iterativos

Existen algunos enfoques para resolver los problemas de los métodos tradicionales, no obstante, los métodos iterativos discutidos a continuación los resuelven de manera mucho más elegante. En vez de almacenar la información global del conjunto de datos, se crea un modelo que aprenderá una iteración a la vez y podrá codificar la probabilidad de una palabra dado su contexto (entendiéndose su contexto como el conjunto de *C* palabras que la rodean). El aprendizaje de este modelo se basa en backpropagation, en el que en cada iteración se calcula el error y se corrige el modelo según corresponda. Los dos modelos descritos a continuación corresponden a los propuestos por Mikolov et al. (2013a). En la Figura 4 se muestra la estructura de estos modelos.

El modelo CBOW se entrena mediante backpropagation como se indicó anteriormente. Se definen los parámetros conocidos del modelo correspondientes a la oración representada según el enfoque de bolsa de palabras común. La palabra a predecir se llamará palabra central y las palabras que la rodean serán el contexto. Se aprenden dos vectores para cada palabra, uno correspondiente a cuando la palabra es la variable a predecir y otro que se utiliza cuando la palabra aparece en el contexto. El modelo skip-gram es conceptualmente similar a CBOW, solo que se intercambian las entradas y salidas. Ahora se utiliza la palabra central para predecir su contexto. La principal diferencia es que se requiere realizar una suposición de independencia condicional fuerte. Es decir, se asume que, dada una palabra central, todas las otras palabras del contexto son independientes.

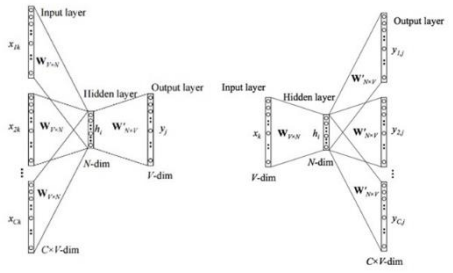

Figura 4: Modelo CBOW (izquierda) y modelo Skip-gram (derecha).

Dado el tamaño del vocabulario, es posible utilizar muestreo negativo para evitar tiempos de entrenamiento muy altos. Para ello, se generan oraciones sintáctica y semánticamente incorrectas para que el modelo logre diferenciar entre las oraciones correctamente formadas y las que no lo son. Esto facilita el proceso de entrenamiento (Mikolov et al. 2013b).

Los trabajos de aprendizaje profundo hacen uso de estas representaciones como entrada en sus redes neuronales profundas. Específicamente, estos enfoques son la base de los métodos utilizados en el estado del arte: word2vec (Mikolov et al. 2013a) y GloVe (Pennington et al. 2014). Ambos métodos son similares y funcionan tomando en cuenta el contexto de la palabra para generar su representación. No obstante, GloVe es un método más poderoso, aunque computacionalmente más costoso. Las descripciones de las siguientes secciones están basadas en los trabajos de estos autores.

Representación word2vec

El método word2vec se basa en una red neuronal de dos capas para procesar el texto. Su entrada es el corpus de texto y su salida es el conjunto de vectores que representan a las palabras en un espacio vectorial semántico. Si bien word2vec no es una red neuronal profunda, convierte el texto en una forma numérica de tal modo que las redes neuronales profundas las puedan entender (Mikolov et al. 2013a).

El principal propósito de word2vec es que permite agrupar vectores de palabras similares en el espacio vectorial, es decir, detecta las similitudes matemáticamente. Ésta suele medirse utilizando la similaridad del coseno. Uno de los resultados más interesantes de la representación en word2vec es que éstas son muy apropiadas para codificar diferentes dimensiones de similitud. Específicamente, es posible realizar pruebas de analogía mediante substracción de vectores. Por otra parte, sus aplicaciones se extienden más allá del procesamiento de texto. Pudiendo ser aplicado a cualquier estructura de estados discreta en las que sea posible estudiar las probabilidades transiciones entre dichos estados (i.e., la probabilidad de co-ocurrencia). Algunos ejemplos de aplicación existentes: representación de genes, código y grafos sociales, entre otras representaciones simbólicas.

El método word2vec es similar a un autoencoder, codificando cada palabra en un vector, pero en vez de entrenar las entradas a través de un proceso de reconstrucción, el método word2vec entrena las palabras con respecto a las palabras que la rodean (i.e. su contexto). Esto se puede realizar de dos formas como se ha visto previamente (CBOW o Skip-gram). Para word2vec se utiliza el segundo, pues produce resultados más precisos en conjuntos de datos más grandes.

Representación GloVe

Se muestra en la Tabla 4 (Pennington et al. 2014) un resumen de alguna de las ventajas (+) y desventajas (-) de las distintas representaciones. Haciéndose distinción con respecto a los métodos basados en conteo (frecuencias) y a los métodos basados en predicción directa (e.g. iterativos). La representación GloVe surge de la idea de combinar las ventajas de los métodos anteriores sin sus desventajas, específicamente busca realizar un uso eficiente de los métodos estadísticos manteniendo el buen rendimiento general y la capacidad de detectar varios patrones.

GloVe es un algoritmo de aprendizaje no supervisado que obtiene representaciones vectoriales para las palabras. El entrenamiento se realiza sobre las estadísticas de co-ocurrencia agregada global entre palabras de un corpus, y las representaciones resultantes muestran interesantes subestructuras lineales en el espacio vectorial de palabras.

La principal intuición de este modelo es que una simple observación de las razones entre las probabilidades de co-ocurrencia de palabras tiene el potencial de encapsular significado.

Tabla 4: Comparación entre modelos de representación de palabras.

	Basados en conteo	**Basados en predicción directa**
Ejemplos	LSA (SVD), HAL (Lund y Burgess 1996), COALS (Rohde et al. 2006), Hellinger-PCA (Lebret y Collobert 2013)	NNLM (Bengio et al 2006), SDHL (Mnih y Hinton 2009), Skip-Gram/CBOW (Mikolov 2013a)
Eficiencia	Entrenamiento rápido. (+)	Entrenamiento escala con el tamaño del corpus. (-)
Propiedades estadísticas	Uso eficiente de estadísticas. (+)	Uso ineficiente de estadísticas. (-)
Rendimiento general	Principalmente utilizado para capturar similaridad de palabras. (-)	Produce un rendimiento mejorado en otras palabras. (+)
Patrones detectables	Entrega una importancia desproporcionada a las cuentas muy altas. (-)	Puede capturar patrones complejos más allá de la similaridad entre palabras. (+)

Al igual que word2vec, GloVe es capaz de capturar interacciones interesantes entre las palabras en el espacio vectorial semántico. Estas interacciones pueden ser a nivel semántico o sintáctico. Estas representaciones encapsulan tanto elementos sintácticos como semánticos, por lo que se hacen especialmente apropiadas para tareas de procesamiento de lenguaje natural.

APRENDIZAJE PROFUNDO

Modelos de lenguajes

Los modelos de lenguaje estadísticos calculan la probabilidad de ocurrencia de las palabras en una oración particular. Estos modelos se basan en la probabilidad de una secuencia de *m* palabras. Como la cantidad de palabras previas a una palabra w_i varía según su posición en el documento, la probabilidad anterior suele estar condicionada sobre una ventana de *n* palabras anteriores en vez de todas las palabras anteriores. Los modelos de lenguaje se basan en este modelo probabilístico (Bengio et al. 2003).

Con el propósito de modelar adecuadamente el lenguaje, Bengio et al. (2003) introducen el primer modelo neuronal utilizado para procesamiento de lenguaje natural. Este modelo permite capturar el contexto de las palabras al aprender una representación distribuida de las palabras (en el sentido de que se encuentra distribuida a lo largo del contexto).

Redes neuronales recurrentes

Las redes neuronales recurrentes son capaces de condicionar el modelo sobre todas las palabras previas en un documento (Mikolov et al. 2010 y 2011). La Figura 5 muestra la arquitectura de una red recurrente, cada rectángulo corresponde a una capa oculta en el paso temporal *t*.

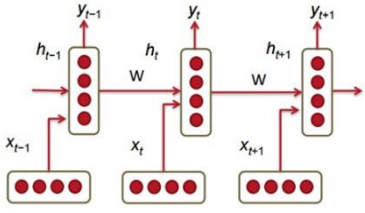

Figura 5: Modelo de redes recurrentes.

Cada una de estas capas tiene una cierta cantidad de neuronas, las cuales realizan una operación lineal sobre sus entradas, seguida por una no lineal. En cada paso temporal, la salida del paso previo en conjunto con la siguiente palabra del documento (en forma de vector x_t), son las entradas de la siguiente capa, produciendo una predicción $\hat{y}$ y características de salida h_t.

Si bien las redes recurrentes son un modelo poderoso, sufren de ciertos problemas con los gradientes al aplicar el algoritmo de backpropagation. Específicamente, sufren de gradientes desvanecientes y la explosión de gradientes. Estas problemáticas pueden ser solucionadas mediante el uso de técnicas más avanzadas. La revisión de estas técnicas escapa del alcance de este documento.

Redes neuronales recursivas

Se discute a continuación un superconjunto de las redes neuronales recurrentes, las redes neuronales recursivas (también abreviadas como RNN). Estos modelos son adecuados para datos que tienen jerarquías anidadas y estructuras recursivas intrínsecas. Esto los hace especialmente útiles para las tareas de NLP, debido a la naturaleza recursiva del lenguaje. Esta sección se encuentra principalmente basada en el trabajo de Socher (2014).

Las reglas sintácticas de un lenguaje son altamente recursivas, esto permite construir modelos que tomen ventajas de esta recursión. Otro beneficio de las RNN es que se pueden analizar oraciones de largo arbitrario con estos modelos, sin tener la problemática de ajustar los datos de entrada a las dimensiones de la red (Bengio et al. 2003).

Uno de los problemas de las representaciones de texto es que varias oraciones que utilizan diferentes palabras y en distintos órdenes tienen significados similares. Esto presenta un problema para los modelos. Existen prácticamente infinitas combinaciones de palabras, por lo que almacenar y entrenar sobre estas es imposible. Además, algunas combinaciones de palabras razonables puede que nunca aparezcan en el conjunto de entrenamiento, así que nunca serían aprendidas.

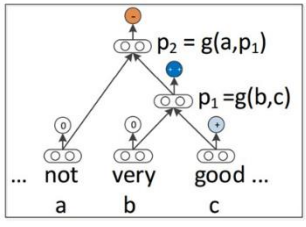

Figura 6: Esquema de aplicación de redes neuronales recursivas al análisis de sentimientos.

Dado esto, se necesita una forma de tomar una oración y sus respectivos vectores de palabras y obtener el vector correspondiente a la oración. Una de las discusiones que existe actualmente en el campo de NLP es si el mismo espacio utilizado para representar las palabras puede ser utilizado para representar las oraciones de largo arbitrario también. Si bien esto parece poco intuitivo, es difícil discutir con los resultados empíricos obtenidos.

El esquema general para aplicar RNN al análisis de sentimientos se muestra en la Figura 6. Los vectores representantes de los padres se calculan siguiendo un enfoque bottom-up utilizando la función de composición g y los vectores que representan cada nodo como atributos de entrada para el clasificador en dicho nodo. La función de composición es la que varía en cada modelo de red neuronal recursiva.

SU-RNN (Socher et al. 2013a)

Una de las limitaciones de las RNN es que se utiliza el mismo W para la composición de todos los elementos sintácticos del lenguaje. Esto no es cierto, pues se esperaría que los verbos, los sujetos y las preposiciones se combinasen de maneras diferentes. Intuitivamente el modelo de las RNN no es lo suficientemente fuerte para expresar todas las complejidades del lenguaje natural.

Se puede remediar esta desventaja mediante la "desvinculación sintáctica" (syntactically untie) de los pesos para estas diferentes tareas. Esto implica que no se usará la misma matriz de pesos W para las distintas categorías de entradas. Con esta adición se obtienen ganancias no triviales de rendimiento, no obstante, se requiere de un parser que detecte las distintas estructuras en las oraciones y realice el etiquetado gramatical de cada fragmento. Para ello se requiere una representación basada en sintaxis discreta (etiquetas gramaticales) y semántica continua (representación del texto). La estructura del árbol de parse se determina mediante el uso de categorías sintácticas de una gramática libre de contexto probabilística (Jelinek et al. 1992).

MV-RNN (Socher et al. 2012)

Se puede ampliar aún más las capacidades del modelo si se incluye no solo un vector para cada palabra, sino también una matriz. Esta matriz permite modelar el efecto de la palabra sobre las otras palabras, por ejemplo, la palabra "muy" tendría asociado un vector $v_{muy} \in \mathbb{R}^d$ y también $V_{muy} \in \mathbb{R}^{d \times d}$. Esto entrega la capacidad expresiva de no solo representar lo que una palabra significa, sino también como modifica a las otras. Este modelo generaliza el caso anterior de SU-RNN, pero en vez de utilizar etiquetas gramaticales para determinar el tipo de efecto que debe tener la palabra, se utilizan las representaciones matriciales para modelar los diferentes efectos.

El modelo MV-RNN es el que posee más parámetros y se basa en entregar el poder a cada palabra de modificar el significado de las palabras circundantes. Requiere de un conjunto de datos grande y su aprendizaje es más lento debido a que cada cálculo requiere tres productos matriciales. Si bien el modelo entrega buenos resultados, tiene

problemas con ciertas construcciones, como las negaciones de frases positivas, las negaciones de frases negativas y las conjunciones contrastivas (e.g. "pero").

RNTN (Socher et al. 2013b)

Las redes neuronales recursivas tensoriales presentan su mayor potencial aplicación en el análisis de sentimientos. Este modelo presenta el mayor éxito en los errores recién expuestos. Esta representación abandona el uso de la matriz para cada palabra y también elimina el uso de la transformación afín. Para componer dos vectores de palabras o vectores de frases, se concatenan los vectores para formar un vector en $\mathbb{R}^{2d}$, pero en vez de utilizar la función afín y luego una función no lineal, se utiliza primero una función cuadrática $h^{(1)} = \tanh(x^T V x + Wx)$.

Donde V es un tensor de tercer orden $\mathbb{R}^{2d \times 2d \times d}$ (un tensor es una generalización del concepto de vector y matriz a dimensiones más altas). Luego, se calculan los cortes $x^T V[i] x \ \forall i \in [1,2,\dots,d]$ del tensor que generan un vector en $\mathbb{R}^d$. A continuación, se agrega Wx y se utiliza una función no lineal. La operación cuadrática permite modelar interacciones entre vectores de palabras sin la necesidad de disponer de matrices que representen a las palabras.

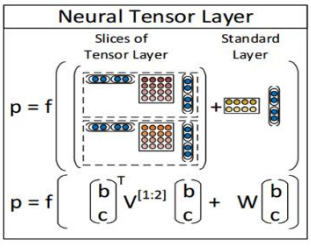

Figura 7: Esquema interno de una RNTN.

Se puede observar un ejemplo de la estructura de la función de composición para las RNTN en la Figura 7. El diagrama muestra los distintos cortes del tensor y la capa estándar del modelo en forma de vectores y matrices, mientras que abajo se observa la función matemática.

El modelo RNTN está basado en la suposición de que existe una única función de composición, pero esta función es extremadamente poderosa. La idea del tensor es permitir interacciones multiplicativas entre los vectores de palabras. Cada corte del tensor tiene la capacidad de capturar diferentes aspectos de la composición.

Acotaciones finales

Todos los resultados expuestos en esta última sección han sido obtenidos del trabajo de Socher (2014). En la Tabla 4 se observan los resultados de clasificación obtenidos con distintos modelos, se observa que el mejor rendimiento (tanto en cinco clases como en dos clases) corresponde al uso de la RNTN.

Tabla 5: Resultados de clasificación para distintos modelos recursivos y línea base.

Modelo / Accuracy	5 clases (--,-,0,+,++)		2 clases (-,+)	
	Nivel nodo	Nivel oración	Nivel nodo	Nivel oración
Unigram	67.2	41.0	82.6	81.8
SVM	64.3	40.7	84.6	89.4
Bigram NB	71.0	41.9	82.7	83.1
VecAvg	73.3	32.7	85.1	80.1
RNN	79.0	43.2	86.1	82.4
MV-RNN	78.7	44.4	86.8	82.9
RNTN	80.7	45.7	87.6	85.4

En función de las discusiones previas, se observa que la principal decisión de diseño para las RNN es la función de composición. Como se ha visto hay distintas motivaciones para cada elección. Por otra parte, la mayoría de los modelos están basados en árboles de sintaxis, no obstante, es posible utilizar modelos basados en árboles de dependencias que tienen la ventaja de capturar la estructura semántica sin depender tanto de la estructura sintáctica. La Tabla 6 presenta un resumen de cada método recursivo.

Tabla 6: Resumen de métodos recursivos.

Método	**Características**
RNN	Son simples, rápidas y se pueden entrenar rápidamente con un algoritmo greedy. No obstante, no son lo suficientemente poderosas para el parsing de oraciones largas.
SU-RNN	Son un modelo muy poderoso, que requiere un conocimiento mínimo de lenguaje y con RNN entrega un buen parser. Las desventajas de este modelo son que requiere de conocimiento lingüístico y el parsing de constitución es muy lento comparado con el parsing de dependencias.
MV-RNN	Tienen un muy buen rendimiento en una variedad de tareas, es un modelo muy poderoso. No obstante, tiene muchos parámetros para ser práctico en la mayoría de los conjuntos de datos.

RNTN	Tienen el mejor rendimiento en análisis de sentimientos. Es un modelo muy poderoso que permite modelar interacciones multiplicativas sin capas ocultas. No tiene demasiadas desventajas, aunque requiere un parser.

Finalmente, se debe indicar que existen otros modelos basados en redes convolucionales aplicadas en procesamiento de lenguaje natural. Las redes recurrentes y recursivas son en realidad un caso particular de estos modelos más generales (Socher 2014).

DISCUSIÓN

La discusión se centrará en responder a las preguntas planteadas en la introducción de esta revisión, las preguntas y sus respectivas respuestas se detallan en la Tabla 7.

Tabla 7: Resumen de preguntas y respuestas.

Pregunta	Respuesta
¿Qué técnicas de clasificación multiclase existen en general?	Se distinguen las técnicas tradicionales (1-vs-all, 1-vs-1, ECOC y descomposición adaptada) y los métodos jerárquicos (DAG y DBT).
¿Es posible aprovechar la estructura ordinal de las clases en este caso?	Considerando que existe un orden natural para los datos, se podría afirmar que la determinación de polaridad multiclase podrían beneficiarse del uso de un método de clasificación ordinal que complemente a los enfoques tradicionales de clasificación.
¿Cuál es el método más adecuado para la determinación de polaridad multiclase?	En función de la naturaleza propia del problema, se podría afirmar que la determinación de polaridad multiclase podría beneficiarse del uso de un método jerárquico o de un método de clasificación ordinal. Posiblemente una combinación que aplique ambos esquemas de manera conjunta entregue los mejores resultados.
¿Qué métodos no han sido aplicados exhaustivamente en la tarea de determinación de polaridad?	Algunos métodos que no han sido ampliamente utilizados para clasificación de sentimientos son las reglas de asociación, lógica difusa, random forests, campos aleatorios condicionales y ontologías. Los métodos como SVM y NB han sido estudiados extensivamente en la literatura por lo que no se consideran de alta prioridad para el desarrollo de un nuevo método.
¿Por qué estos métodos podrían ser útiles?	Estos métodos poseen una serie de ventajas que no han sido exploradas ampliamente en la literatura de determinación de polaridad. Se cree que utilizando las propiedades de éstos métodos o estrategias inspiradas en ellos se podría desarrollar un mejor clasificador multiclase.
¿Qué falencias presentan en el caso multiclase?	Todos los modelos presentan una baja en cuanto a la clasificación multiclase a nivel de oración. En general, todos los modelos estudiados no son lo suficientemente poderosos para modelar el caso multiclase por sí mismos. De aquí surge la motivación para aplicar combinaciones adecuadas de estos.
¿Qué tipo de representación del texto es la más adecuada para este problema?	En función de lo estudiado se recomienda utilizar la representación word2vec o GloVe, pues permiten capturar varias dimensiones de similaridad sintáctica y semántica, además de otros aspectos importantes. Estas mismas representaciones se pueden utilizar no solo para las palabras, sino para las oraciones.
¿Qué técnicas de preprocesamiento son necesarias para obtener esta representación?	Las técnicas de preprocesamiento requeridas no difieren de los requerimientos estándar del procesamiento de lenguaje natural, principalmente la tokenización y el análisis estadístico de co-ocurrencias.
¿Qué información semántico-sintáctica almacenan estas representaciones?	Los métodos almacenan varias dimensiones de información semántica y sintáctica. Principalmente, son capaces de realizar analogías mediante simples relaciones lineales y agrupan palabras similares en regiones cercanas. También son capaces de representar oraciones completas en el mismo espacio mediante el uso de funciones de composición.
¿Qué técnicas de clasificación basada en métodos recursivos profundos han sido aplicadas con éxito en determinación de polaridad?	Todas las variantes de redes neuronales recursivas y recurrentes han sido aplicadas con éxito al problema la determinación de polaridad. El modelo con más éxito corresponde a RNTN.
¿Qué ventajas y desventajas poseen las diferentes técnicas?	Una de las desventajas de algunos modelos es el requerimiento de un parser auxiliar, la necesidad de grandes conjuntos de datos y el tiempo de entrenamiento.

CONCLUSIÓN

Se han concretado los objetivos propuestos en esta revisión, específicamente, se han respondido las preguntas planteadas inicialmente en este artículo. Se destaca principalmente los avances en representación del texto mediante espacios vectoriales semánticos en el campo del procesamiento de lenguaje natural. Este desarrollo se ha visto ligado fuertemente con el avance del campo del aprendizaje profundo, utilizando y complementado las técnicas que surgen en dicha área.

Teniendo ahora una visión más clara de los métodos utilizados en el estado del arte, tanto en lo que respecta a enfoques tradicionales y de aprendizaje profundo, el desafío pendiente consiste en utilizar esto como base para el diseño de nuevos métodos y nuevos enfoques. Se propone como trabajo futuro el desarrollo de un método híbrido y/o recursivo que aproveche las diferentes características de los métodos expuestos en este artículo.

REFERENCIAS

[1] Lorena, A. C., De Carvalho, A. C., & Gama, J. M. (2008). A review on the combination of binary classifiers in multiclass problems. Artificial Intelligence Review, 30(1-4), 19-37.

[2] Dietterich, T. G., & Bakiri, G. (1995). Solving multiclass learning problems via error-correcting output codes. Journal of artificial intelligence research, 263-286.

[3] Kotsiantis, S. B., & Pintelas, P. E. (2004). A cost sensitive technique for ordinal classification problems. In Methods and applications of artificial intelligence (pp. 220-229). Springer Berlin Heidelberg.

[4] Frank, E., & Hall, M. (2001). A simple approach to ordinal classification (pp. 145-156). Springer Berlin Heidelberg.

[5] Wang, D., Zhai, J., Zhu, H., & Wang, X. (2014, July). An Improved Approach to Ordinal Classification. In Machine Learning and Cybernetics (pp. 33-42). Springer Berlin Heidelberg.

[6] Pang, B., & Lee, L. (2005, June). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (pp. 115-124). Association for Computational Linguistics.

[7] Baccianella, S., Esuli, A., & Sebastiani, F. (2010, May). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. InLREC (Vol. 10, pp. 2200-2204).

[8] Ravi, K., & Ravi, V. (2015). A survey on opinion mining and sentiment analysis: tasks, approaches and applications. Knowledge-Based Systems, 89, 14-46.

[9] Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal, 5(4), 1093-1113.

[10] Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. ACM SIGMOD Record, 22(2), 207-216.

[11] Liu B. Hsu W. Ma Yiming (1998, August). Integrating classification and association rule mining. In Proceedings of the fourth international conference on knowledge discovery and data mining.

[12] Klir, G., & Yuan, B. (1995). Fuzzy sets and fuzzy logic (Vol. 4). New Jersey: Prentice hall.

[13] Li, S. T., & Tsai, F. C. (2013). A fuzzy conceptualization model for text mining with application in opinion polarity classification. Knowledge-Based Systems,39, 23-33.

[14] Ho, T. K. (1995, August). Random decision forests. In Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on (Vol. 1, pp. 278-282). IEEE.

[15] Ho, T. K. (1998). The random subspace method for constructing decision forests. Pattern Analysis and Machine Intelligence, IEEE Transactions on,20(8), 832-844.

[16] Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1). Springer, Berlin: Springer series in statistics.

[17] Sutton, C., & McCallum, A. (2006). An introduction to conditional random fields for relational learning. Introduction to statistical relational learning, 93-128.

[18] Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., & Torr, P. H. (2015). Conditional random fields as recurrent neural networks.

InProceedings of the IEEE International Conference on Computer Vision (pp. 1529-1537).

[19] Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing?. International journal of human-computer studies, 43(5), 907-928.

[20] Turney, P. D., & Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. Journal of artificial intelligence research, 37(1), 141-188.

[21] Liu, B. (2007). Web data mining: exploring hyperlinks, contents, and usage data. Springer Science & Business Media.

[22] Dumais, S. T. (2004). Latent semantic analysis. Annual review of information science and technology, 38(1), 188-230.

[23] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013a). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[24] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

[25] Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global Vectors for Word Representation. In EMNLP (Vol. 14, pp. 1532-1543).

[26] Lund, K., & Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. Behavior Research Methods, Instruments, & Computers, 28(2), 203-208.

[27] Rohde, D. L., Gonnerman, L. M., & Plaut, D. C. (2006). An improved model of semantic similarity based on lexical co-occurrence. Communications of the ACM, 8, 627-633.

[28] Lebret, R., & Collobert, R. (2013). Word emdeddings through hellinger PCA. arXiv preprint arXiv:1312.5542.

[29] Bengio, Y., Schwenk, H., Senécal, J. S., Morin, F., & Gauvain, J. L. (2006). Neural probabilistic language models. In Innovations in Machine Learning (pp. 137-186). Springer Berlin Heidelberg.

[30] Mnih, A., & Hinton, G. E. (2009). A scalable hierarchical distributed language model. In Advances in neural information processing systems (pp. 1081-1088).

[31] Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010, September). Recurrent neural network based language model. In INTERSPEECH (Vol. 2, p. 3).

[32] Mikolov, T., Kombrink, S., Burget, L., Černocký, J. H., & Khudanpur, S. (2011, May). Extensions of recurrent neural network language model. In Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on (pp. 5528-5531). IEEE.

[33] Irsoy, O., & Cardie, C. (2014, October). Opinion Mining with Deep Recurrent Neural Networks. In EMNLP (pp. 720-728).

[34] Socher, R. (2014). Recursive Deep Learning for Natural Language Processing and Computer Vision (Doctoral dissertation, Stanford University).

[35] Socher, R., Huval, B., Manning, C. D., & Ng, A. Y. (2012, July). Semantic compositionality through recursive matrix-vector spaces. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (pp. 1201-1211). Association for Computational Linguistics.

[36] Socher, R., Bauer, J., Manning, C. D., & Ng, A. Y. (2013, June). Parsing with Compositional Vector Grammars. In ACL (1) (pp. 455-465).

[39] Jelinek, F., Lafferty, J. D., & Mercer, R. L. (1992). Basic methods of probabilistic context free grammars (pp. 345- 360). Springer Berlin Heidelberg.

[38] Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013, October). Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the conference on empirical methods in natural language processing (EMNLP) (Vol. 1631, p. 1642).